

Manuel d'installation pour plateforme d'hébergement Ruby on Rails sur DEDIBOX

Table des matières

I) Installation d'un environnement Ruby On Rails.....	4
1) Se connecter en tant que root sur la plateforme.....	4
2) Installer Ruby et ses composants :.....	4
3) Télécharger et installer Gems.....	4
4) Installer Rails.....	5
5) Installation d'apache.....	5
6) Installation de Mysql et ses librairies.....	6
7) Installer PHP4.....	6
8) Installation de phpmyadmin.....	6
II) Configuration de la plateforme.....	7
1) Configurer Mysql.....	7
a) Commandes ajout de mot de passe root.....	7
b) Résultats des commandes précédentes.....	7
c) Ajouter l'utilisateur Rails a mysql.....	7
2) Configurer Apache.....	8
a) Activer le mode fastcgi.....	8
b) Activer le module rewrite.....	8
c) Créer deux dossiers pour les logs des vhosts.....	8
d) Activer le module rewrite.....	8
e) Il faut modifier l'htaccess dans tous les projets /public/.htaccess.....	8
III) Installation spécifique dédié pour l'hébergement mutualisé.....	9
1) Installation de bind9 (daemon: named).....	9
2) Installation de ProFTP (daemon: proftpd).....	9
3) Installation de quota de disque (Pas de daemon).....	9
4) Installation de Postfix (daemon: postfix + pickup + qmgr).....	10
5) Installation de la QOS.....	10
IV) Configuration des outils pour l'hébergement mutualisé.....	11
1) Configuration de bind9.....	11
a) Modifier le fichier de configuration /etc/bind/named.conf.....	11
b) Créer le fichier /etc/bind/db.b4os.fr.....	11
c) Créer le fichier /etc/bind/db.b4os.fr.inv.....	12
2) Configuration de ProFTP	12
3) Configuration des quotas.....	14
4) Configuration de Postfix.....	15
5) Configuration de la QOS.....	16
6) Configuration d'un firewall grace à IPTABLES.....	16
V) Configuration avancé.....	20
1) Postfix.....	20
2) Installation postfix TLS-SASL.....	22
3) Script de creation d'utilisateur.....	24
4) Script de suppression d'utilisateur	28
5) Configuraion de Logrotate	30
a) dans /etc/webalizer/cron.daily/all.....	30
b) dans /etc/webalizer/cron.weekly/all.....	31
6) Installer le Vhost d'accueil + phpmyadmin.....	35
VI) Installation d'outils pour les performances et tests.....	36
1) Pour les stats, installer Webalizer (logiciel utilise par Free) :.....	36
2) Installation de ntop pour verifier la BP de tous les projets :.....	36

I) Installation d'un environnement Ruby On Rails

1) Se connecter en tant que root sur la plateforme

dedibox:/\$ su - root

Pour installer les applications il faut absolument avoir les droits necessaire sur le systeme.

2) Installer Ruby et ses composants :

dedibox:/# apt-get install ruby libzlib-ruby rdoc irb

Reading Package Lists... Done

Building Dependency Tree... Done

The following extra packages will be installed:

irb1.8 libreadline-ruby1.8 libreadline4 libruby1.8 rdoc1.8 ruby1.8

Suggested packages:

ruby1.8-examples ri1.8

The following NEW packages will be installed:

irb irb1.8 libreadline-ruby1.8 libreadline4 libruby1.8 libzlib-ruby rdoc rdoc1.8 ruby ruby1.8

0 upgraded, 10 newly installed, 0 to remove and 0 not upgraded.

Need to get 2202kB of archives.

After unpacking 7643kB of additional disk space will be used.

3) Télécharger et installer Gems

<http://rubyforge.org/projects/rubygems/>

(dedibox:/# wget <http://rubyforge.org/frs/download.php/5207/rubygems-0.8.11.tgz>

le lien etait mort pour l'install alors copie sur site perso)

dedibox:/# wget <http://pierre.fauquembergue.free.fr/rubygems-0.8.11.tgz>

dedibox:/# tar zxvf rubygems-0.8.11.tgz

dedibox:/# cd rubygems-0.8.11

dedibox:/# ruby setup.rb all

Successfully built RubyGem

Name: sources

Version: 0.0.1

File: sources-0.0.1.gem

4) Installer Rails

dedibox:/# gem install rails --include-dependencies

Attempting local installation of 'rails'
Local gem file not found: rails.gem*
Attempting remote installation of 'rails'
Updating Gem source index for: http://gems.rubyforge.org
Successfully installed rails-1.1.2
Successfully installed rake-0.7.1
Successfully installed activesupport-1.3.1
Successfully installed activerecord-1.14.2
Successfully installed actionpack-1.12.1
Successfully installed actionmailer-1.2.1
Successfully installed actionwebservice-1.1.2
Installing RDoc documentation for rake-0.7.1...
Installing RDoc documentation for activesupport-1.3.1...
Installing RDoc documentation for activerecord-1.14.2...
Installing RDoc documentation for actionpack-1.12.1...
Installing RDoc documentation for actionmailer-1.2.1...
Installing RDoc documentation for actionwebservice-1.1.2...

Rails est installé, nous devons maintenant installer un serveur Web (apache2)

5) Installation d'apache

dedibox:/# apt-get install apache2 libapache2-mod-fcgid libfcgi-ruby1.8

Reading Package Lists... Done
Building Dependency Tree... Done
The following extra packages will be installed:
apache2-common apache2-mpm-worker apache2-utils libapr0 libexpat1 libfcgi0 libldap2
libmagic1 libsasl2 mime-support openssl ssl-cert
Suggested packages:
apache2-doc lynx www-browser file ca-certificates
Recommended packages:
libsasl2-modules
The following NEW packages will be installed:
apache2 apache2-common apache2-mpm-worker apache2-utils libapache2-mod-fcgid
libapr0 libexpat1 libfcgi-ruby1.8 libfcgi0 libldap2 libmagic1 libsasl2 mime-support
openssl ssl-cert
0 upgraded, 15 newly installed, 0 to remove and 0 not upgraded.
Need to get 3226kB of archives.
After unpacking 10.1MB of additional disk space will be used.

6) Installation de Mysql et ses librairies

```
dedibox:/# apt-get install mysql-server-4.1 libmysqlclient14 libmysql-ruby1.8
```

Reading Package Lists... Done

Building Dependency Tree... Done

The following extra packages will be installed:

libdbd-mysql-perl libdbi-perl libmysqlclient12 libnet-daemon-perl libplrpc-perl mysql-client-4.1 mysql-common-4.1

Suggested packages:

dbshell libcompress-zlib-perl mysql-doc-4.1

Recommended packages:

libmysqlclient14-dev

The following NEW packages will be installed:

libdbd-mysql-perl libdbi-perl libmysql-ruby1.8 libmysqlclient12 libmysqlclient14 libnet-daemon-perl libplrpc-perl mysql-client-4.1 mysql-common-4.1 mysql-server-4.1

7) Installer PHP4

```
dedibox:/# apt-get install php4 php4-mysql php4-cli libapache2-mod-php4
```

Reading Package Lists... Done

Building Dependency Tree... Done

The following extra packages will be installed:

apache-common apache2-mpm-prefork libapache-mod-php4 libbz2-1.0 libkrb53 libncursesw5 libzip-0-12 lynx php4-common

Suggested packages:

apache apache-ssl apache-perl php4-pear libbz2-dev bzip2 krb5-doc krb5-user

The following packages will be REMOVED:

apache2-mpm-worker

The following NEW packages will be installed:

apache-common apache2-mpm-prefork libapache-mod-php4 libapache2-mod-php4 libbz2-1.0 libkrb53 libncursesw5 libzip-0-12 lynx php4 php4-cli php4-common php4-mysql

Pour vérifier que PHP fonctionne correctement avec apache, il faut créer le fichier /var/www/apache2-default/test.php :

```
<? phpinfo(); ?>
```

8) Installation de phpmyadmin

```
dedibox# apt-get install phpmyadmin
```

1) Which web server would you like to reconfigure automatically?
apache2

2) Would you restart apache2 ?
yes

II) Configuration de la plateforme

1) Configurer Mysql

a) Commandes ajout de mot de passe root

```
dedibox:/# mysql -u root
mysql> UPDATE mysql.user SET Password =
PASSWORD('nouveau_mot_de_passe') WHERE User = 'root';
mysql> FLUSH PRIVILEGES;
```

b) Résultats des commandes précédentes

```
dedibox:/# mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8 to server version: 4.1.11-Debian_4sarge2-log

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> UPDATE mysql.user SET Password =
      PASSWORD('nouveau_mot_de_passe') WHERE User =
'root';
Query OK, 2 rows affected (0.00 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)
```

c) Ajouter l'utilisateur Rails a mysql

```
dedibox:/# mysql -u root -p
mysql> GRANT ALL PRIVILEGES ON *.* TO 'rails'@'localhost'
      IDENTIFIED BY 'un_mot_de_passe' WITH GRANT OPTION;
```

```
Résultat :
mysql> GRANT ALL PRIVILEGES ON *.* TO 'rails'@'localhost' IDENTIFIED BY
      'un_mot_de_passe' WITH GRANT OPTION;
Query OK, 0 rows affected (0.00 sec)
```

2) Configurer Apache

a) Activer le mode fastcgi

```
dedibox:/# a2enmod fcgid
```

Module fcgid installed; run /etc/init.d/apache2 force-reload to enable.

```
dedibox:/# /etc/init.d/apache2 restart
```

Forcing reload of web server: Apache2.

```
dedibox:/# gem install fcgi
```

b) Activer le module rewrite

```
dedibox:/# a2enmod rewrite
```

```
dedibox:/# /etc/init.d/apache2 reload
```

c) Créer deux dossiers pour les logs des vhosts

```
dedibox:/# /var/log/apache2/error
```

```
dedibox:/# /var/log/apache2/access
```

d) Activer le module rewrite

Après les 2 lignes suivantes dans /etc/apache2/apache2.conf

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
```

```
LogFormat "%h %l %u %t \"%r\" %>s %b" common
```

Ajouter cette ligne

```
LogFormat "%v %h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combinedvhosth
```

e) Il faut modifier l'htaccess dans tous les projets /public/.htaccess

On commente

```
AddHandler fastcgi-script .fcgi
```

Et on remplace par

```
AddHandler fcgid-script .fcgi
```

On peut aussi changer la règle de réécriture par :

```
RewriteRule ^(.*)$ dispatch.fcgi \[QSA,L\]
```

III) Installation spécifique dédié pour l'hébergement mutualisé

1) Installation de bind9 (daemon: named)

dedibox:/# apt-get install bind9

*Reading Package Lists... Done
Building Dependency Tree... Done
The following extra packages will be installed:
libdns16 libisc7 libisccc0 libisccfg0 liblwres1
Suggested packages:
dnsutils bind9-doc
The following NEW packages will be installed:
bind9 libdns16 libisc7 libisccc0 libisccfg0 liblwres1
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
Need to get 1141kB of archives.
After unpacking 2720kB of additional disk space will be used.*

2) Installation de ProFTP (daemon: proftpd)

dedibox:/# apt-get install proftpd

*Reading Package Lists... Done
Building Dependency Tree... Done
The following extra packages will be installed:
proftpd-common ucf
Suggested packages:
proftpd-doc
Recommended packages:
debconf-utils
The following NEW packages will be installed:
proftpd proftpd-common ucf
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 598kB of archives.
After unpacking 1589kB of additional disk space will be used.*

Choisir l'option 'standalone' pour de meilleures performances

3) Installation de quota de disque (Pas de daemon)

dedibox:/# apt-get install quota

*Reading Package Lists... Done
Building Dependency Tree... Done
The following NEW packages will be installed:
quota
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 419kB of archives.*

After unpacking 1122kB of additional disk space will be used.

Répondre '**Yes**' pour envoyer un rappel si l'utilisateur dépasse son quota.
Puis donner l'adresse email à contacter en cas de problème.

4) Installation de Postfix (daemon: postfix + pickup + qmgr)

dedibox:/# apt-get install postfix

```
Reading Package Lists... Done
Building Dependency Tree... Done
Suggested packages:
  procmail postfix-mysql postfix-pgsql postfix-ldap postfix-pcre
Recommended packages:
  resolvconf
The following packages will be REMOVED:
  exim4 exim4-base exim4-config exim4-daemon-light
The following NEW packages will be installed:
  postfix
0 upgraded, 1 newly installed, 4 to remove and 0 not upgraded.
Need to get 802kB of archives.
After unpacking 1118kB disk space will be freed.
```

Questions relatives à l'installation :

- a) General type of configuration => '**Internet Site**'
- b) Where should mail for root go => '**kinder**'
- c) Mail name (after @) => '**b4os.fr**'
- d) Other destinations to accept mail for? => '**b4os.fr, sd-2156.dedibox.fr, dev4ror.b4os.fr, localhost.dedibox.fr, localhost**'
- e) Force synchronous updates on mail queue? => '**Yes**' (plus lent, mais sans perte de mail)

Lors de l'installation de Postfix, apt-get doit supprimer automatiquement Exim4.

5) Installation de la QOS

***** A FAIRE *****

Pour l'instant, QOS avec iptables :

```
# Priorisation des connexions ftp et ssh
iptables -A PREROUTING -t mangle -p tcp --sport ssh -j TOS --set-tos Minimize-Delay
iptables -A PREROUTING -t mangle -p tcp --sport ftp -j TOS --set-tos Minimize-Delay
# On donne un maximum de débit aux transferts ftp, peu importe la latence
iptables -A PREROUTING -t mangle -p tcp --sport ftp-data -j TOS --set-tos Maximize-Throughput
```

***** A FAIRE *****

IV) Configuration des outils pour l'hébergement mutualisé

1) Configuration de bind9

a) Modifier le fichier de configuration /etc/bind/named.conf

Ajouter ces lignes en fin de fichier :

```
//Ma zone a moi !
zone "b4os.fr" {
    type master;
    file "/etc/bind/db.b4os.fr";
};

//Zone inverse a moi
zone "151.21.191.88.in-addr.arpa" {
    type master;
    file "/etc/bind/db.b4os.fr.inv";
};
```

b) Creer le fichier /etc/bind/db.b4os.fr

Puis ajouter ces lignes :

```
;
; BIND reverse data file for local loopback interface
;
$TTL 604800
@ IN SOA dev4ror.b4os.fr. root.localhost. (
    1 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL
;
@ IN NS dev4ror.b4os.fr.
1.0.0 IN PTR localhost.

dev4ror IN A 88.191.21.151
www IN CNAME dev4ror.b4os.fr.

; ** Les lignes suivantes d\xc3\xa9finissent la table entre les noms et les IP
www IN A 88.191.21.151
shop IN A 88.191.21.151
```

c) Creer le fichier /etc/bind/db.b4os.fr.inv

```
;  
; BIND reverse data file for local loopback interface  
;  
$TTL 604800  
@ IN SOA ror.zieute.com. root.zieute.com. (  
    1  
    604800  
    86400  
    2419200  
    604800 )  
  
@ IN NS dev4ror.b4os.fr.  
  
1 IN PTR dev4ror.b4os.fr.  
2 IN PTR shop.b4os.fr.  
3 IN PTR www.b4os.fr.
```

Pour tester que le demon fonctionne bien, redemarrer le service comme suit :

dedibox:/# /etc/init.d/bind9 restart

Vérifier les logs :

dedibox:/# tail - 30 /var/log/syslog

2) Configuration de ProFTP

En premier temps, securiser son ftp

dedibox:/# cp /etc/passwd /etc/ftpusers

Ensuite il faut editer le fichier de config

dedibox:/# vi /etc/proftpd.conf

```
#  
# /etc/proftpd.conf -- This is a basic ProFTPD configuration file.  
# To really apply changes reload proftpd after modifications.  
#
```

```
ServerName          "FTP B4OS"  
ServerType          standalone  
DeferWelcome        off
```

```
MultilineRFC2228    on  
DefaultServer       on  
ShowSymlinks        on
```

```
TimeoutNoTransfer    600  
TimeoutStalled       600  
TimeoutIdle          1200
```

```
ServerIdent on "Authentication en cours..."  
DisplayLogin         welcome.msg  
DisplayFirstChdir    .message
```

```

ListOptions          "-l"

DenyFilter           \ *.* /

#
# Auth
#
AccessGrantMsg "Connexion Reussie..."
AccessDenyMsg "Authentication Incorrecte..."
RequireValidShell    off
#
# Log
#
ServerLog /var/log/proftpd.log
TransferLog /var/log/proftpd-transfert.log


# Uncomment this if you are using NIS or LDAP to retrieve passwords:
#PersistentPasswd     off


# Uncomment this if you would use TLS module:
#TLSEngine            on


# Uncomment this if you would use quota module:
#Quotas               on


# Uncomment this if you would use ratio module:
#Ratios               on


# Port 21 is the standard FTP port.
Port                 21382


# To prevent DoS attacks, set the maximum number of child processes
# to 30. If you need to allow more than 30 concurrent connections
# at once, simply increase this value. Note that this ONLY works
# in standalone mode, in inetd mode you should use an inetd server
# that allows you to limit maximum number of processes per service
# (such as xinetd)
MaxInstances         30


# Set the user and group that the server normally runs at.
User                 nobody
Group                nogroup


# Umask 022 is a good standard umask to prevent new files and dirs
# (second parm) from being group and world writable.
Umask                022 022
# Normally, we want files to be overwriteable.
AllowOverwrite       on


#Connexion standard
Defaultroot ~


#Utilisateur autorises
<Limit LOGIN>
AllowUser kinder
DenyAll
</Limit>

```

3) Configuration des quotas

il faut monter la partition qui sera l'objet de quotas avec l'option désirée :

usrquota : active les quotas utilisateurs

grpquota : active les quotas groupes

Nous choisirons ici l'option usrquota; pour l'appliquer à la partition /home, ouvrez /etc/fstab :

```
dedibox:/# vi /etc/fstab
```

et ajouter "usrquota" après les options existantes :

```
/dev/sda3    /home  ext3  defaults,usrquota    0    2
```

créer le fichier aquota.user à la racine de la partition à laquelle vous voulez appliquer les quotas, puis le rendre exécutable.

```
dedibox:/# touch /home/aquota.user
```

```
dedibox:/# chmod 600 /home/aquota.user
```

Il faut alors remonter les partitions en question :

```
dedibox:/# mount -o remount /home
```

Vérifiez ensuite le système de quota à l'aide de la commande :

```
dedibox:/# quotacheck -auvg
```

Attendre la fin du traitement de la commande :

```
quotacheck: WARNING - Quotafile /home/aquota.user was probably truncated. Can't save quota settings...
```

```
quotacheck: Scanning /dev/sda3 [/home] quotacheck: Old group file not found. Usage will not be subtracted.
```

```
done
```

```
quotacheck: Checked 5 directories and 6 files
```

Pour activer les quotas, utilisez la commande :

```
dedibox:/# quotaon -a
```

Il faut finalement appliquer les quotas pour chaque utilisateur

```
dedibox:/# edquota -u user
```

S'ouvre alors le fichier qui définit les quotas de l'utilisateur; il est constitué de trois colonnes, par exemple :

```
Disk quotas for user test (uid 1001):
```

```
Filesystem blocks soft hard inodes soft$
```

```
/dev/hda3 12 0 100000 3 0$
```

Pour kinder :

```
Disk quotas for user kinder (uid 1000):
```

```
Filesystem blocks quota limit grace files quota limit grace
```

```
/dev/sda3    16 900000 1000000          4    0    0
```

Filesystem comporte les Systèmes de fichiers concernés par les quotas;
Blocks est l'espace utilisé
Soft est la limite douce appliquée aux blocks : l'utilisateur sera seulement prévenu,
Hard est la limite dure appliquées aux blocks : l'utilisateur ne pourra pas dépasser cette limite.

Les inodes constituent des "cartes d'identité" pour chaque fichier.

Ici, aucune limite douce n'est fixée (La valeur de "/dev/hda3" pour soft est "0"),
tandis que l'utilisateur ne pourra pas occuper plus de 100Mo dans son espace personnel
(la valeur de "/dev/hda3" pour "hard" est 100000)

Pour Prévenir l'utilisateur lorsqu'il a atteint sa limite soft, modifier le fichier
/etc/warnquota.conf

4) Configuration de Postfix

Détail de la configuration du fichier « /etc/postfix/main.cf »

Définition du nom du serveur et du domaine :

myhostname = mail.mondomaine.com

mydomain = mondomaine.com

Lire les requêtes depuis toutes les interfaces réseau :

inet_interfaces = all

Quel domaine afficher dans le courrier sortant :

myorigin = \$mydomain

De quels domaines recevoir le courrier :

mydestination = \$myhostname localhost.\$mydomain \$mydomain

De quels clients relayer le courrier :

mynetworks = 192.168.0/8, 10.2.2/8, 10.2.3/8, 127.0.0.0/8

Gestion des alias et de la re-direction des mails :

alias_maps = hash:/etc/postfix/aliases

Adresse du relais SMTP pour les courrier non traités par le serveur :

relayhost = smtp.free.fr

Ré-écriture de l'adresse de l'expéditeur (ligne à ajouter -> facultatif) :

sender_canonical_maps = hash:/etc/postfix/canonical

ATTENTION : A chaque modification de ce fichier ou des tables ci-dessous, il faut recharger la configuration :

dedibox: /# /etc/rc.d/init.d/postfix reload

5) Configuration de la QOS

***** A FAIRE *****

Pour l'instant gere avec iptables

***** A FAIRE *****

6) Configuration d'un firewall grace à IPTABLES

Pour finir l'installation, on installe un firewall afin de sécuriser la plateforme.

Mettre le fichier ci dessous dans /etc/init.d/firewall.sh

Et lui ajoindre les droits necessaire a l'execution :

dedibox:/# chmod ugo+x /etc/init.d/firewall.sh

***** DEBUT DU FIREWALL *****

#!/bin/sh

Firewall dev4ror

firewall_start() {

echo "[Demarrage du firewall]"

CHARGEMENT DES MODULES

echo "[Chargement des modules]"

modprobe ip_tables

modprobe iptable_filter

modprobe iptable_mangle

modprobe iptable_nat

REGLES PAR DEFAULT

echo "[Initialisation de la table filter]"

iptables -F #Elimine toutes les regles

iptables -X #Suppression d'une chaine vide

echo "[Politique par défaut de la table filter]"

On ignore tout ce qui entre ou transite par la passerelle

iptables -P INPUT DROP

iptables -P FORWARD DROP

On accepte, ce qui sort

iptables -P OUTPUT ACCEPT

Pour éviter les mauvaises surprises, on va

autoriser l'accès à la loopback, c'est vital !

iptables -A INPUT -i lo -j ACCEPT

iptables -A OUTPUT -o lo -j ACCEPT

LOCAL-INTERNET

```

echo "[On autorise les clients à accéder à internet ]"

#On créé une nouvelle chaîne "local-internet"
iptables -N local-internet #Nouvelle chaine

iptables -A local-internet -m state --state NEW -i !eth1 -j ACCEPT

iptables -A local-internet -m state --state ESTABLISHED,RELATED -j ACCEPT

# On termine en indiquant que les connections appartenant à "local-internet"
# accèdent à internet de manière transparente.
iptables -A INPUT -j local-internet
iptables -A FORWARD -j local-internet

##### LES TABLES NAT ET MANGLE #####

echo "[Initialisation des tables nat et mangle]"

iptables -t nat -F #Elimine les regles
iptables -t nat -X #Supprime la chaine vide
iptables -t nat -P PREROUTING ACCEPT #Chgt de politique pour
serveurs
iptables -t nat -P POSTROUTING ACCEPT #Chgt de politique pour
masquerade
iptables -t nat -P OUTPUT ACCEPT #Pout la passerelle elle meme

iptables -t mangle -F
iptables -t mangle -X
iptables -t mangle -P PREROUTING ACCEPT #Optimise la connexion
iptables -t mangle -P OUTPUT ACCEPT

##### PAS DE SPOOFING #####

echo "[Pas de spoofing]"
if [ -e /proc/sys/net/ipv4/conf/all/rp_filter ] ; then
for filtre in /proc/sys/net/ipv4/conf/*/rp_filter
do
echo 1 > $filtre
done
fi

##### PAS DE SYNFOOD #####

echo "[Pas de synflood]"
if [ -e /proc/sys/net/ipv4/tcp_syncookies ] ; then
echo 1 > /proc/sys/net/ipv4/tcp_syncookies
fi

##### PAS DE PING #####

# commentez ces 6 lignes, si vous autorisez les pings sur votre passerelle
echo "[Pas ping]"
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
if [ -e /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses ] ; then
echo 1 > /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses
fi

```

```

##### CONNECTION SSH #####

echo "[Connection SSH Disponible ]"

iptables -A INPUT -p tcp --dport 22333 -j ACCEPT
#Priorité aux connexions ssh
iptables -A PREROUTING -t mangle -p tcp --sport 22333 -j TOS --set-tos
Minimize-Delay

##### CONNECTION FTP #####

echo "[Connection FTP Disponible ]"

iptables -A INPUT -p tcp --dport 21382 -j ACCEPT
#Priorité aux connexions ftp
iptables -A PREROUTING -t mangle -p tcp --sport 21382 -j TOS --set-tos
Minimize-Delay

##### AUTORISATION DU FORWARD 80 #####

echo "[Serveur apache visible depuis internet ]"

iptables -A INPUT -p tcp --dport 80 -j ACCEPT

##### AUTORISATION DU PORT 22080 #####

echo "[Serveur NTOP accessible ]"

iptables -A INPUT -p tcp --dport 22080 -j ACCEPT

##### AUTORISATION SMTP 25 #####

echo "[Serveur mail autorise ]"

iptables -A INPUT -p tcp --dport 25 -j ACCEPT
#iptables -A OUTPUT -o eth0 -p tcp --sport 1024: --dport 25 -m state --state !
INVALID -j ACCEPT
#iptables -A INPUT -i eth0 -p tcp --sport 25 --dport 1024: -m state --state
RELATED,ESTABLISHED -j ACCEPT

##### AUTORISATION DNS 53 #####

echo "[Serveur DNS fonctionnel ]"

iptables -A INPUT -p udp --dport 53 -j ACCEPT
#iptables -A OUTPUT -o eth0 -p udp --sport 1024: --dport 53 -m state --state !
INVALID -j ACCEPT
#iptables -A INPUT -i eth0 -p udp --sport 53 --dport 1024: -m state --state
RELATED,ESTABLISHED -j ACCEPT

##### FIN DU FIREWALL #####

echo "[firewall activé !]"

```

```

}

firewall_stop() {

    iptables -F
    iptables -X
    iptables -P INPUT ACCEPT
    iptables -P FORWARD ACCEPT
    iptables -P OUTPUT ACCEPT

    iptables -t nat -F
    iptables -t nat -X
    iptables -t nat -P PREROUTING ACCEPT
    iptables -t nat -P POSTROUTING ACCEPT
    iptables -t nat -P OUTPUT ACCEPT

    iptables -t mangle -F
    iptables -t mangle -X
    iptables -t mangle -P PREROUTING ACCEPT
    iptables -t mangle -P OUTPUT ACCEPT

    echo "[firewall descativé! ]"
}

firewall_restart() {
    firewall_stop
    sleep 2
    firewall_start
}

case "$1" in
    'start')
        firewall_start
        ;;
    'stop')
        firewall_stop
        ;;
    'restart')
        firewall_restart
        ;;
    'status')
        iptables -L
        iptables -t nat -L
        iptables -t mangle -L
        ;;
    *)
        echo "Usage: firewall {start|stop|restart|status}"
esac

```

***** FIN DU FIREWALL *****

Puis ajouter un lien symbolique dans rc.d pour qu'il soit lancé a chaque demarrage :

dedibox: /# ln -s /etc/init.d/firewall.sh /etc/rc2.d/S12firewall

V) Configuration avancé

1) Postfix

Création du fichier des aliases « /etc/postfix/aliases »

Le fichier des alias permet de re-diriger une adresse vers une autre boîte (même externe) ou d'écrire une même adresse sous plusieurs formes.

La création de la table des aliases s'effectue avec la commande spécifique « postalias » et en mettant le signe « : » comme séparateur de colonne alors que les autres tables (canonical, transport...) sont créées avec la commande « postmap » sans le signe « : » comme séparateur :

Éditer le fichier « /etc/postfix/aliases » et saisir les informations sous la forme suivante :

```
root :    pglinux
pglinux :    tony.galmiche@pg.com
tg :    tony.galmiche@pg.com
```

Créer la table à partir du fichier texte :

dedibox:/# postalias /etc/postfix/aliases

ATTENTION : Après chaque modification de ce fichier, il faut recréer la table et recharger le fichier de configuration :

dedibox:/# postalias /etc/postfix/aliases
dedibox:/# /etc/init.d/postfix reload

Création du fichier de ré-écriture de l'adresse de l'expéditeur.

L'utilisation de ce fichier est facultatif pour le fonctionnement de « Postfix »

Éditer le fichier /etc/postfix/canonical et saisir les informations sous la forme suivante :

```
tg    tony.galmiche@pg.com
```

Créer la table à partir du fichier texte :

dedibox:/# postmap /etc/postfix/canonical

ATTENTION : Après chaque modification de ce fichier, il faut recréer la table et recharger le fichier de configuration :

dedibox:/# postmap /etc/postfix/canonical
dedibox:/# /etc/init.d/postfix reload

Création du fichier des modes de transport.

Ce fichier permet de spécifier un mode de transport différent de « SMTP » ou un relais SMTP spécifique en fonction de chaque domaine.

Éditer le fichier « /etc/postfix/transport » et saisir les informations sous la forme suivante :

```
domaine1.com    local:
domaine2.com    smtp : 10.2.2.1
```

Créer la table à partir du fichier texte :

dedibox:/# postmap /etc/postfix/transport

ATTENTION : Après chaque modification de ce fichier, il faut recréer la table et recharger le fichier de configuration :

dedibox:/# postmap /etc/postfix/transport

dedibox:/# /etc/rc.d/init.d/postfix reload

Consultation des logs

Lire les logs des messages envoyés :

dedibox:/# tail -f /var/log/mail/info

Compter les messages deferred par Postfix :

dedibox:/# find /var/spool/postfix/deferred -type f | wc

***** FICHIER POSTFIX main.cf *****

See /usr/share/postfix/main.cf.dist for a commented, more complete version

smtpd_banner = Serveur de Dev B4OS // ESMTP
biff = no

Repertoires d'installation de Postfix:
command_directory = /usr/sbin
daemon_directory = /usr/lib/postfix
program_directory = /usr/lib/postfix
queue_directory = /var/spool/postfix

#Mise en place des Permissions
setgid_group = postdrop
mail_owner = postfix

#Configuration machine
myhostname = dev4ror.b4os.fr
mydomain = b4os.fr
masquerade_domains = b4os.fr
mynetworks = 127.0.0.0/8

#Parametres d'envoi des courriers
myorigin = \$mydomain
default_transport = smtp

alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
mydestination = \$myhostname, localhost.\$mydomain, \$mydomain, localhost
#relayhost = smtp.free.fr
#mailbox_command = procmail -a "\$EXTENSION"
#mailbox_size_limit = 0
#recipient_delimiter = +
inet_interfaces = all

2) Installation postfix TLS-SASL

dedibox: /# apt-get install postfix-tls libsasl2 libsasl2-modules sasl2-bin openssl

Générons le certificat SSL valable 1 an pour le TLS de postfix :

dedibox: /# cd /etc/postfix

**dedibox: /# openssl req -new -outform PEM -out /etc/postfix/smtpd.cert
-newkey rsa:2048 -nodes -keyout /etc/postfix/smtpd.key -keyform
PEM -days 365 -x509**

Répondez aux quelques questions demandées, la plus importante étant : "Common Name (eg, YOUR name) :", où il faudra indiquer le FQDN de votre machine (machine.domaine.org).

Country Name (2 letter code) [AU]: => FR
State or Province Name (full name) [Some-State] : FRANCE
Locality Name (eg, city) : BORDEAUX
Organization Name (eg, company) [Internet Widgits Pty Ltd] : B4OS
Organizational Unit Name (eg, section) : SSL
Common Name (eg, YOUR name) :dev4ror.b4os.fr
Email Address : pierre.fauquembergue@b4os.fr

Corrigeons maintenant les droits sur le fichier de la clé pour éviter qu'un utilisateur mal intentionné vienne la lire.

dedibox: /# chmod o= /etc/postfix/smtpd.key

Ajoutez l'utilisateur postfix au groupe sasl, ceci permettra à postfix d'aller dialoguer avec saslauthd grâce au socket unix /var/run/saslauthd/mux.

dedibox: /# adduser postfix sasl

Passons maintenant à la configuration de saslauthd :
Editez le fichier /etc/default/saslauthd :
Décommentez START=yes en enlevant le # en début de ligne.
Spécifiez : MECHANISMS="pam shadow".

Pour que postfix puisse lire le fichier /etc/shadow :

dedibox: /# adduser postfix shadow

Ensuite créez le répertoire /etc/postfix/sasl :

**dedibox: /# mkdir /etc/postfix/sasl && chown root:postfix
/etc/postfix/sasl**

Et créez y le fichier smtpd.conf contenant :

pwcheck_method: saslauthd
mech_list: plain login
allow_plaintext: true

On indique maintenant à PAM que l'on désire utiliser une authentification de type UNIX . Ceci se fait en creant le fichier /etc/pam.d/smtp :

```
#
# The PAM configuration file for the smtp service
#
# The standard Unix authentication modules, used with
# NIS (man nsswitch) as well as normal /etc/passwd and
# /etc/shadow entries.
@include common-auth
@include common-account
@include common-session
@include common-password
```

Il faut maintenant modifier le fichier /etc/postfix/master.cf en decomentant cette ligne :

```
smtps    inet  n       -       -       smtpd -o smtpd_tls_wrappermode=yes -o
smtpd_sasl_auth_enable=yes
```

Et finissons avec la configuration de postfix :
Editons le fichier /etc/postfix/main.cf, et ajoutons :

```
smtpd_recipient_restrictions = permit_mynetworks, permit_sasl_authenticated,
reject_unauth_destination
smtpd_sasl_auth_enable = yes
broken_sasl_auth_clients = yes
smtpd_use_tls = yes
smtpd_tls_cert_file = /etc/postfix/smtpd.cert
smtpd_tls_key_file = /etc/postfix/smtpd.key
smtpd_sasl_local_domain = $myhostname
smtpd_sasl_security_options = noanonymous
```

```
* on crée un répertoire /var/run dans /var/spool/postfix
dedibox:/# mkdir -p /var/spool/postfix/var/run

* on déplace /var/run/saslauthd dans /var/spool/postfix/var/run
dedibox:/# mv /var/run/saslauthd /var/spool/postfix/var/run

* on crée un lien symbolique de /var/spool/postfix/var/run/saslauthd vers /var/run
dedibox:/# ln -s /var/spool/postfix/var/run/saslauthd
/var/run/saslauthd
```

Enfin, redémarrons le tout :

```
dedibox:/# /etc/init.d/postfix restart
dedibox:/# /etc/init.d/saslauthd restart
```

Voilà ! Il ne vous reste plus qu'à configurer votre client mail favori pour se connecter en TLS avec authentification "PLAIN", y indiquer votre login et votre mot de passe.

3) Script de creation d'utilisateur

```
#!/bin/bash

if [ $UID -eq 0 ]
then

if [ ! $3 ]
then
echo "Usage: `basename $0` nom_utilisateur mot_de_passe nom_du_projet"
exit 1
fi

#Variable intialisation

user="$1"
pass="$2"
project="$3"

#Creation de l'utilisateur
/usr/sbin/useradd -g railers -s /bin/false $user

#Met a jour le mot de passe
echo $user:$pass | /usr/sbin/chpasswd

if grep -E "^$user:" /etc/passwd > /dev/null ;
then
echo "Utilisateur $user cree avec succes"
else
echo "L'utilisateur $user n'a pas pu etre cree"
exit 1
fi

#Creation de son repertoire d'accueil
/bin/mkdir /home/$user
/bin/chown $user /home/$user

if [ -d /home/$user ]
then
echo "Le repertoire /home/$user est cree"
else
echo "Le repertoire /home/$user n'a pas pu etre cree"
exit 1
fi

#Creation de son compte MYSQL
sql="CREATE DATABASE $project;"
sql="$sql GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP"
sql="$sql ON $project . * TO '$user'@%' IDENTIFIED BY '$pass'"

#fichier temporaire
echo "$sql" > temp.sql

#lancement de la requete
echo "Compte MYSQL créer"
/usr/bin/mysql -u rails -p6vdelap1 mysql < temp.sql

#effacer le fichier temporaire
rm temp.sql
```

```

#Cree le compte FTP
echo " Creation du compte FTP "
cp /etc/proftpd.conf /tmp/proftpd.txt
sed -e 132i"AllowUser $user" /tmp/proftpd.txt > /tmp/proftpd.tmp
cp /tmp/proftpd.tmp /etc/proftpd.conf

#On relance le service ftp
echo " Relance le service FTP "
/etc/init.d/proftpd reload

#Creation du quota de disque
echo " Creation du quota de disque "
/usr/sbin/edquota -p kinder $user

# Creation du dossier de logs apache et webalizer
/bin/mkdir -m 777 /var/log/apache2/error/$project
/bin/mkdir -m 777 /var/log/apache2/access/$project
/bin/mkdir -m 777 /var/www/webalizer/$project

#Creer un hote virtuel pour APACHE
echo " Cree un hote virtuel "

text="<VirtualHost *:80> \n"
text="$text  ServerName $project \n"
text="$text  ServerAlias $project.b4os.fr \n"
text="$text  DocumentRoot /var/www/$project/public/ \n"
text="$text  ErrorLog /var/log/apache2/error.log \n"
text="$text  ErrorLog /var/log/apache2/error/$project/error.log \n"
text="$text  CustomLog /var/log/apache2/access.log combined \n"
text="$text  CustomLog /var/log/apache2/access.vhost.log combinedvhost \n"
text="$text  CustomLog /var/log/apache2/access/$project/access.log combined
\n"
text="$text  CustomLog /var/log/apache2/access/$project/access.vhost.log
combinedvhost \n"
text="$text  ServerPath /var/www/$project/public/ \n"
text="$text  SetEnv RAILS_ENV production \n"
text="$text <Directory /var/www/$project/public/> \n"
text="$text    Options ExecCGI FollowSymLinks \n"
text="$text    AddHandler fcgid-script .fcgi \n"
text="$text    AllowOverride all \n"
text="$text    Order allow,deny \n"
text="$text    Allow from all \n"
text="$text </Directory> \n"
text="$text</VirtualHost> \n"

echo -e "$text" > /etc/apache2/sites-available/$project

#Ajout dans Logrotate
echo " Ajout dans Logrotate "

insert="\n#Projet $project \n"
insert="$insert  /var/log/apache2/access/$project/*.log { \n"
insert="$insert    weekly \n"
insert="$insert    missingok \n"
insert="$insert    rotate 52 \n"
insert="$insert    compress \n"
insert="$insert    delaycompress \n"
insert="$insert    notifempty \n"
insert="$insert    create 640 root adm \n"
insert="$insert    sharedscripts \n"
insert="$insert    postrotate \n"
insert="$insert    /etc/init.d/apache2 reload > /dev/null \n"

```

```

insert="$insert endscript \n"
insert="$insert } \n \n"

echo -e "$insert" >> /etc/logrotate.d/apache2

#Ajout du fichier webalizer conf pour le projet
#Ajout dans webalizer daily
echo " Ajout dans webalizer daily "

daily="#!/bin/sh"
daily="$daily\n#Powered by B4OS \n"
daily="$daily\nWEBALIZER_BIN=/usr/bin/webalizer"
daily="$daily\nWEBALIZER_CONF=/etc/webalizer/conf/$project.conf \n"
daily="$daily\n# See if the webalizer binary and config file exists"
daily="$daily\n[ -f \${WEBALIZER_BIN} ] || exit 0"
daily="$daily\n[ -f \${WEBALIZER_CONF} ] || exit 0 \n"
daily="$daily\n# Figure out non-rotated logfile"
daily="$daily\nnonrotatedlog=\`egrep '^LogFile' \${WEBALIZER_CONF} | \"
daily=\"$daily\nsed -e 's/[[:space:]]\+/ /' | \"
daily=\"$daily\ncut -d ' ' -f2 | \"
daily=\"$daily\nsed -e 's/[.[:digit:]][\.gz]*\$/ /' | \"
daily=\"$daily\n# Can we read it? \"
daily=\"$daily\n[ -r \${nonrotatedlog} ] || exit 0 \n"
daily=\"$daily\n# Check for empty logfile"
daily=\"$daily\nlogsz=\`echo \${nonrotatedlog} | \"
daily=\"$daily\nsed -e 's/[[:space:]]\+/ /' | \"
daily=\"$daily\ncut -d ' ' -f2 | \"
daily=\"$daily\nxargs ls -l | \"
daily=\"$daily\nsed -e 's/[[:space:]]\+/ /g' | \"
daily=\"$daily\ncut -d ' ' -f5 | \"
daily=\"$daily\n[ \${logsz} -gt 0 ] || exit 0 \n"
daily=\"$daily\n# Run webalizer quietly"
daily=\"$daily\n\${WEBALIZER_BIN} -c \${WEBALIZER_CONF} -q \${nonrotatedlog}
\n"

daily="$daily\n# Exit with webalizer's exit code"
daily="$daily\nexit \${?} "

echo -e "$daily" > /etc/webalizer/cron.daily/$project

#Ajout dans webalizer weekly
echo " Ajout dans webalizer weekly "

weekly="#!/bin/sh"
weekly="$weekly\n#Powered by B4OS \n"
weekly="$weekly\nWEBALIZER_BIN=/usr/bin/webalizer"
weekly="$weekly\nWEBALIZER_CONF=/etc/webalizer/conf/$project.conf \n"
weekly="$weekly\n# See if the webalizer binary and config file exists"
weekly="$weekly\n[ -f \${WEBALIZER_BIN} ] || exit 0"
weekly="$weekly\n[ -f \${WEBALIZER_CONF} ] || exit 0 \n"
weekly="$weekly\n# Figure out non-rotated logfile"
weekly="$weekly\n\${log}=\`egrep '^LogFile' \${WEBALIZER_CONF} | \"
weekly=\"$weekly\nsed -e 's/[[:space:]]\+/ /' | \"
weekly=\"$weekly\ncut -d ' ' -f2 | \"
weekly=\"$weekly\nsed -e 's/[.[:digit:]][\.gz]*\$/ /' | \"
weekly=\"$weekly\n# Can we read it? \"
weekly=\"$weekly\n[ -r \${\${log}} ] || exit 0 \"
weekly=\"$weekly\n# Check for empty logfile \"
weekly=\"$weekly\nlogsz=\`echo \${\${log}} | \"
weekly=\"$weekly\nsed -e 's/[[:space:]]\+/ /' | \"
weekly=\"$weekly\ncut -d ' ' -f2 | \"
weekly=\"$weekly\nxargs ls -l | \"
weekly=\"$weekly\nsed -e 's/[[:space:]]\+/ /g' | \"

```

```
weekly="$weekly\ncut -d ' ' -f5 \` \"
weekly="$weekly\n[ \${logsz} -gt 0 ] || exit 0 \n\"
weekly="$weekly\n# Run webalizer quietly\"
weekly="$weekly\n\${WEBALIZER_BIN} -c \${WEBALIZER_CONF} -q \${\${log}} \n\"
weekly="$weekly\n# Exit with webalizer's exit code\"
weekly="$weekly\nexit \${?} \"
```

```
echo -e \"$weekly\" > /etc/webalizer/cron.weekly/$project
```

```
#Rendre ces fichiers executable
/bin/chmod 755 /etc/webalizer/cron.daily/$project
/bin/chmod 755 /etc/webalizer/cron.weekly/$project
```

```
#Creer le fichier de conf
```

```
conf=\"# Webalizer configuration file\"
conf=\"$conf\n#For $project.b4os.fr domain\"
conf=\"$conf\n#Powered by B4OS \n\"
conf=\"$conf\nLogFile /var/log/apache2/access/$project/access.log.1\"
conf=\"$conf\nOutputDir /var/www/webalizer/$project\"
conf=\"$conf\nReportTitle Statistiques pour\"
conf=\"$conf\nHostName $project.b4os.fr \n\"
conf=\"$conf\n#Referers a cacher\"
conf=\"$conf\nHideReferrer *.b4os.fr\"
```

```
echo -e \"$conf\" > /etc/webalizer/conf/$project.conf
```

```
#Cree le repertoire du site
echo \" Cree un lien symbolique entre le repertoire de l'utilisateur \"
echo \" Et le chemin d'acces du repertoire public apache \"
ln -s /home/$user /var/www/$project
```

```
#Active l'hote virtuel
echo \"Active l'hote virtuel\"
/usr/sbin/a2ensite $project
```

```
#Relance APACHE
echo \" Relance le service Apache \"
/etc/init.d/apache2 reload
```

```
#Enregistrement DNS dans la zone locale Zieute
echo \" Enregistre le projet dans la zone DNS \"
echo -e \"$project IN A 88.191.21.151\" >> /etc/bind/db.b4os.fr
```

```
#Recharge le DNS
echo \" Recharge la zone DNS \"
/etc/init.d/bind9 reload
```

```
else
echo \"Vous n'avez pas les droits necessaires\"
fi
```

4) Script de suppression d'utilisateur

```
#!/bin/bash

if [ $UID -eq 0 ]
then

    if [ ! $2 ]
    then
        echo "Usage: `basename $0` nom_utilisateur nom_du_projet"
        exit 1
    fi

    #Variable intialisation
    user="$1"
    project="$2"

    #Suppression de l'utilisateur
    /usr/sbin/userdel $user

    #Suppression de son repertoire d'accueil
    /bin/rm /var/www/$project
    /bin/rm -Rf /home/$user

    if [ -d /home/$user ]
    then
        echo "Le repertoire /home/$user est toujours existant"
        exit 1
    else
        echo "Le repertoire /home/$user est supprimé"
    fi

    #Suppression de son compte MYSQL
    sql="DROP DATABASE $project;"
    sql="$sql REVOKE ALL PRIVILEGES, GRANT OPTION FROM $user;"
    sql="$sql DROP USER $user;"

    #fichier temporaire
    echo "$sql" > temp.sql

    #lancement de la requete
    echo "Suppression du compte MYSQL"
    /usr/bin/mysql -u rails -p6vdelap1 mysql < temp.sql

    #effacer le fichier temporaire
    /bin/rm temp.sql

    #Suppression du compte FTP
    echo " Suppression du compte FTP "

    cptftp=`/bin/sed -n "/^AllowUser $user/=" /etc/proftpd.conf`
    if [ ! $cptftp ]
    then
        echo "Utilisateur non inscrit dans Proftpd"
    else
        /bin/sed "s/$cptftp d/" /etc/proftpd.conf > /tmp/cptftp
        cp /tmp/cptftp /etc/proftpd.conf
    fi

    #On relance le service ftp
```

```

echo " Relance le service FTP "
/etc/init.d/proftpd reload

# Suppression du dossier de logs apache et webalizer
/bin/rm -Rf /var/log/apache2/error/$project
/bin/rm -Rf /var/log/apache2/access/$project
/bin/rm -Rf /var/www/webalizer/$project

#Suppression dans Logrotate
echo " Suppression dans Logrotate "

ligne=`/bin/sed -n "/Projet $project/=" /etc/logrotate.d/apache2`
if [ ! $ligne ]
then
echo "Projet non inscrit dans logrotate"
else
/bin/sed "$ligne,$(($ligne+13))d" /etc/logrotate.d/apache2 > /tmp/logrotate
cp /tmp/logrotate /etc/logrotate.d/apache2
fi

#Suppression dans webalizer daily
/bin/rm -f /etc/webalizer/cron.daily/$project

#Suppression dans webalizer weekly
/bin/rm -f /etc/webalizer/cron.weekly/$project

#Suppression du fichier de conf
/bin/rm -f /etc/webalizer/conf/$project.conf

#Desactive l'hote virtuel
echo "desactive l'hote virtuel"
/usr/sbin/a2dissite $project

#Suppression de l'hote virtuel
echo " Suppression de l'hote virtuel "
/bin/rm -f /etc/apache2/sites-available/$project

#Relance APACHE
echo " Relance le service Apache "
/etc/init.d/apache2 reload

#Suppression DNS dans la zone locale b4os
echo " Suppression du projet dans la zone DNS "

nomdns=`/bin/sed -n "/^$project/=" /etc/bind/db.b4os.fr`
if [ ! $nomdns ]
then
echo "Projet non inscrit dans la zone dns"
else
/bin/sed "$nomdns d" /etc/bind/db.b4os.fr > /tmp/nomdns
cp /tmp/nomdns /etc/bind/db.b4os.fr
fi

#Recharge le DNS
echo " Recharge la zone DNS "
/etc/init.d/bind9 reload

else
echo "Vous n'avez pas les droits necessaires"
fi

```

5) Configuraion de Logrotate

Cree 3 dossiers a la racine de /etc/webalizer/

- * conf
- * cron.daily
- * cron.weekly

Prenez le fichier /etc/cron.daily/webalizer et sauvegardez le (/etc/webalizer/webalizer.cron.daily.old).

Modifiez ensuite le fichier /etc/cron.daily/webalizer de cette façon :

```
#!/bin/sh  
run-parts --report /etc/webalizer/cron.daily
```

Idem pour le fichier /etc/cron.weekly/webalizer, sauvegardez le puis modifiez le :

```
#!/bin/sh  
run-parts --report /etc/webalizer/cron.weekly
```

On cree deux fichiers pour l'analyse de toutes les zones :

a) dans /etc/webalizer/cron.daily/all

```
#!/bin/sh  
#Powered by B4OS  
  
WEBALIZER_BIN=/usr/bin/webalizer  
WEBALIZER_CONF=/etc/webalizer/conf/all.conf  
  
# See if the webalizer binary and config file exists  
# if not, exit without warning to prevent daily mails  
[ -f ${WEBALIZER_BIN} ] || exit 0  
[ -f ${WEBALIZER_CONF} ] || exit 0  
  
# Figure out non-rotated logfile  
nonrotatedlog=`egrep '^LogFile' $WEBALIZER_CONF | \  
sed -e 's/[[:space:]]\+/ /' | \  
cut -d ' ' -f2 | \  
sed -e 's/\.[[:digit:]]*\.[gz]*$/ /' ``  
  
# Can we read it?  
[ -r ${nonrotatedlog} ] || exit 0  
  
# Check for empty logfile  
logsz=`echo ${nonrotatedlog} | \  
sed -e 's/[[:space:]]\+/ /' | \  
cut -d ' ' -f2 | \  
xargs ls -l | \  
sed -e 's/[[:space:]]\+/ /g' | \  
cut -d ' ' -f5`  
[ $logsz -gt 0 ] || exit 0  
  
# Run webalizer quietly  
${WEBALIZER_BIN} -c ${WEBALIZER_CONF} -q ${nonrotatedlog}  
  
# Exit with webalizer's exit code  
exit $?
```

b) dans /etc/webalizer/cron.weekly/all

```
#!/bin/sh
#Powered by B4OS

WEBALIZER_BIN=/usr/bin/webalizer
WEBALIZER_CONF=/etc/webalizer/conf/all.conf

# See if the webalizer binary and config file exists
# if not, exit without warning to prevent daily mails
[ -f ${WEBALIZER_BIN} ] || exit 0
[ -f ${WEBALIZER_CONF} ] || exit 0

# Figure out non-rotated logfile
${log}=`egrep '^LogFile' $WEBALIZER_CONF | \
sed -e 's/[[:space:]]\+/ /' | \
cut -d ' ' -f2 | \
sed -e 's/\.[[:digit:]]*\.[[:digit:]]*/'`

# Can we read it?
[ -r ${log} ] || exit 0

# Check for empty logfile
logsz=`echo ${log} | \
sed -e 's/[[:space:]]\+/ /' | \
cut -d ' ' -f2 | \
xargs ls -l | \
sed -e 's/[[:space:]]\+/ /g' | \
cut -d ' ' -f5`
[ $logsz -gt 0 ] || exit 0

# Run webalizer quietly
${WEBALIZER_BIN} -c ${WEBALIZER_CONF} -q ${log}

# Exit with webalizer's exit code
exit $?
```

Creation du dossier /var/www/webalizer/all

```
dedibox# mkdir -m 777 /var/www/webalizer/all
```

et Un fichier pour le fichier de conf : /etc/webalizer/conf/all.conf

Le fichier de configuration pour TOUS les domaines

```
# Webalizer configuration file
# For all domain about b4os.fr
# Powered by B4OS

LogFile /var/log/apache2/access.log.1
OutputDir /var/www/webalizer/all
ReportTitle Statistiques pour Tous les domaines B4OS
HostName dev4ror.b4os.fr

# Referers à cacher

HideReferrer dev4ror.b4os.fr

# End of configuration file... Have a nice day!
```

La manipulation étant exactement la même pour /etc/webalizer/cron.weekly, je vous laisse l'accomplir, reprenez par contre le fichier /etc/cron.weekly/webalizer que vous aviez sauvé.

Le fichier /etc/webalizer.conf

Nous allons de créer autant de fichier de configuration que de domaines, mais il y a un fichier principal à créer, qui contient les directives nécessaires à tous les domaines/vhosts, c'est le fichier /etc/webalizer.conf.

```
# Sample Webalizer configuration file
# Powered by B4OS

LogType web
HistoryName webalizer.hist
Incremental yes
IncrementalName webalizer.current
HTMLExtension html
UseHTTPS no

HTMLHead <META name="keywords" content="webalizer">
HTMLBody <BODY BGCOLOR="#E8E8E8" TEXT="#000000" LINK="#0000FF"
VLINK="#FF0000">
HTMLPost <BR CLEAR="all">
HTMLTail B4OS !
HTMLEnd </BODY></HTML>

Quiet no
ReallyQuiet no
TimeMe no
GMTTime no
Debug no
IgnoreHist no

HourlyGraph yes
HourlyStats yes
CountryGraph yes
DailyGraph yes
DailyStats yes

PageType *.htm
PageType *.html
PageType *.php
PageType *.wml

GraphLegend yes
GraphLines 2
VisitTimeout 1800

#SearchEngine est déjà défini dans le code pour pas mal de moteurs.

AllAgents yes
AllReferrers yes
AllSites yes
AllURLs yes
AllSearchStr yes
AllUsers yes

TopSites 20
TopKSites 20
TopURLs 20
TopKURLs 20
```

*TopReferrers 20
TopAgents 15
TopCountries 20
TopEntry 20
TopExit 20
TopSearch 15
TopUsers 10*

===== SITES =====

*HideSite *dev4ror*

*GroupSite *aol.com AOL
GroupSite *wanadoo.fr Wanadoo
GroupSite *proxad.net Free
GroupSite *tiscali.fr Tiscali
GroupSite *club-internet.fr Club-Internet
GroupSite *nerim.net Nerim
GroupSite *mindspring.com EarthLink
GroupSite *verizon.net Verizon Online
GroupSite *t-dialin.net T-Online
GroupSite *comcast.net Comcast
GroupSite *optonline.com Optimum Online
GroupSite *sympatico.ca Sympatico.ca
GroupSite *rogers.com Rogers
GroupSite *skynet.be Skynet
GroupSite *noos.fr Noos
GroupSite *noos.net Noos*

===== REFERRERS =====

*# A cacher suivants les domaines
#HideReferrer moomoocamp.net
#HideReferrer powah.net
#HideReferrer yaz0r.net
#HideReferrer darkmoo.net
#HideReferrer dhm2000.org*

*# Toujours cachés, pour ceux la.
HideReferrer dev4ror/
HideReferrer Direct Request*

*# Groupons les moteurs de recherche
GroupReferrer yahoo.com/ Yahoo!
GroupReferrer yahoo.fr/ Yahoo!
GroupReferrer http://search.yahoo Yahoo!
GroupReferrer excite.com/ Excite
GroupReferrer infoseek.com/ InfoSeek
GroupReferrer webcrawler.com/ WebCrawler
GroupReferrer google.com/ Google
GroupReferrer google.fr/ Google
GroupReferrer http://www.google Google*

*# Cachons tout ça, puisqu'il sont groupés juste au dessus
HideReferrer yahoo.com/
HideReferrer yahoo.fr/
HideReferrer http://search.yahoo
HideReferrer google.com/
HideReferrer google.fr/
HideReferrer http://www.google
HideReferrer excite.com/
HideReferrer infoseek.com/*

HideReferrer webcrawler.com/

===== URLS =====

***HideURL *.gif
HideURL *.GIF
HideURL *.jpg
HideURL *.JPG
HideURL *.jpeg
HideURL *.JPEG
HideURL *.png
HideURL *.PNG
HideURL *.bmp
HideURL *.BMP
HideURL *.ra***

GroupURL /cgi-bin/*

===== AGENTS =====

Créons des groupes pour les browsers, et ne cachons pas les détails.

GroupAgent MSIE Microsoft Internet Explorer

GroupAgent Mozilla

GroupAgent Lynx

GroupAgent Winamp Nullsoft Winamp

GroupAgent Googlebot

GroupAgent Konqueror

GroupAgent Opera

GroupAgent FAST-WebCrawler

GroupAgent GetRight

GroupAgent Wget

GroupAgent Links

GroupAgent FrontPage

GroupAgent Scooter

GroupAgent Mosaic

GroupAgent DA* Download Accelerator

#HideAgent MSIE

#HideAgent Mozilla

#HideAgent Lynx

#HideAgent Winamp

#HideAgent Googlebot

#HideAgent Konqueror

#HideAgent Opera

#HideAgent FAST-WebCrawler

#HideAgent GetRight

#HideAgent Wget

#HideAgent Links

#HideAgent FrontPage

#HideAgent Scooter

#HideAgent Mosaic

#HideAgent DA*

MangleAgents 4

Et pour finir quelques options bidon

IgnoreSite localhost

IgnoreReferrer localhost

GroupShading yes

GroupHighlight yes

6) Installer le Vhost d'accueil + phpmyadmin

Cree le fichier suivant : /etc/apache2/sites-available/accueil

NameVirtualHost *:80

```
<VirtualHost *:80>
    ServerName dev4ror
    ServerAlias dev4ror.b4os.fr
    ErrorLog /var/log/apache2/www_error.log
    CustomLog /var/log/apache2/www_access.log combined
    Redirect permanent / http://www.b4os.fr
</VirtualHost>

<VirtualHost *:80>
    ServerName phpmyadmin
    ServerAlias phpmyadmin.b4os.fr
    ErrorLog /var/log/apache2/phpmyadmin_error.log
    CustomLog /var/log/apache2/phpmyadmin_access.log combined
    DocumentRoot /var/www/phpmyadmin/
</VirtualHost>
```

Cree le fichier suivant : /etc/apache2/sites-available/www

```
<VirtualHost *:80>
    ServerName www
    ServerAlias www.b4os.fr
    ErrorLog /var/log/apache2/www_error.log
    CustomLog /var/log/apache2/www_access.log combined
    DocumentRoot /var/www/www/public/
    ServerPath /var/www/www/public/
    <Directory /var/www/www/public/>
        Options ExecCGI FollowSymLinks
        AddHandler fcgid-script .fcgi
        AllowOverride all
        Order allow,deny
        Allow from all
    </Directory>
</VirtualHost>
```

telecharger le projet ici

```
dedibox# cd /var/www/
dedibox# wget
http://pierre.fauquembergue.free.fr/divers/railswww.tar.gz
dedibox# tar zxvf railswww.tar.gz
```

Puis activer les vhosts :

```
dedibox# a2ensite www
dedibox# a2ensite accueil
```

VI) Installation d'outils pour les performances et tests

1) Pour les stats, installer Webalizer (logiciel utilise par Free) :

dedibox:/# apt-get install webalizer

```
Reading Package Lists... Done
Building Dependency Tree... Done
The following extra packages will be installed:
  libfreetype6 libgd2-noxpm libjpeg62 libpng12-0
Suggested packages:
  libfreetype6-dev libgd-tools
The following NEW packages will be installed:
  libfreetype6 libgd2-noxpm libjpeg62 libpng12-0 webalizer
0 upgraded, 5 newly installed, 0 to remove and 0 not upgraded.
Need to get 1059kB of archives.
After unpacking 3232kB of additional disk space will be used.
```

Questions a l'installation :

Which directory should webalizer put the output in? => **'/var/www/webalizer'**
Enter the title of the reports webalizer will generate. => **'Statistiques pour '**
What is the filename of the rotated webserver log? => **'/var/log/apache/access.log.1'**

Assurez vous que votre serveur apache génère des log en mode : combined, dans votre fichier httpd.conf
CustomLog /logs/access_log combined

Ensuite modifier le fichier /etc/webalizer.conf en fonctions des besoins.
Puis pour enregistrer cette configuration faite :

dedibox:/# \$WEBALIZER/bin/webalizer -c /etc/webalizer.conf

2) Installation de ntop pour verifier la BP de tous les projets :

dedibox:/# apt-get install ntop

```
Reading Package Lists... Done
Building Dependency Tree... Done
The following NEW packages will be installed:
  ntop
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 2402kB of archives.
After unpacking 6525kB of additional disk space will be used.
```

Questions a l'installation :

Which interfaces should ntop listen on? => **'eth0'**
Which is the name of the user to run the ntop daemon as ? => **'ntop'**

Installation sans aucun probleme sur Debian Sarge ...

Bon alors ntop c'est sympa mais sa doc elle est vraiment pas RTFM friendly, elle devrait vraiment préciser ces deux/trois trucs pour éviter de galérer :

si on lance le daemon :

dedibox: /# /etc/init.d/ntop start

Ca marche pas, le daemon plante en quelques secondes ...

- Il faut lancer ntop à la main au moins une fois, pour qu'il vous demande un mot de passe admin, après ca ira mieux :

dedibox: /# ntop -u ntop

- u ntop c'est pour dire à ntop d'utiliser l'utilisateur système "ntop", ntop ne doit pas tourner en root

- Après une petite modification de /etc/init.d/ntop :

#!/bin/sh

DAEMON="/usr/sbin/ntop"

NAME="ntop"

DESC="network top daemon"

INIT="/etc/default/ntop"

HOMEDIR="/var/lib/ntop"

LOGDIR="/var/log/ntop/"

test -f \$DAEMON || exit 0

test -f \$INIT || exit 0

. \$INIT

case "\$1" in

start)

echo -n "Starting \$DESC: "

**start-stop-daemon --start --quiet --name \$NAME --exec \$DAEMON -- **

**-d -L -u \$NAME -P \$HOMEDIR --skip-version-check **

**-a /var/log/ntop/access.log -i "\$INTERFACES" **

**-w 22080 **

-O \$LOGDIR \$GETOPT

if ps xa | grep -v grep | grep /usr/sbin/ntop > /dev/null ; then

echo ntop

else

si on (re)lance le daemon :

dedibox: /# /etc/init.d/ntop start

on doit pouvoir accéder à la page : <http://adresseIPmachine:22080>